Design Support and Prototyping

[Preece Chap 23-28; Hix Chap 2,9,11; Neilsen Chap 8-9; Newman Chap 8-9]

Standards

Task: Select

Description: The user specifies a selection from a set of alternatives on an interactive display. The implication is that there is something (an entity) that is selectable, and that there is more than one possible selection. The alternatives might consist of a set of commands, such as items in a menu, or a collection of entities, such as graphic objects.

Task Phrase: <specify> <entity> (from <set of alternatives>)

Properties:
<specify>: none

<entity>: type as identified in conceptual model (for example, command vs. graphic object selection may be two different tasks)

<set of alternatives>: ordered vs. not ordered; range(if variable set); size (if fixed set); may be implied by context if there is a 'currently selected object'

Recommended Techniques:
Keyboard: entity label type-in
Mouse: cursor match

See also Apple standard for window scrolling [Included]

Guidelines

[From Hix & Hartson]

User-Centered Design
    Practice user-centered design
    Know the user
    Involve the user via participatory design
    Prevent user errors
    Optimize user operations
    Keep the locus of control with the user
    Help the user get started with the system
System Model
    Give the user a mental model of the system, based on user tasks
Consistency and Simplicity
    Be consistent
    Keep it simple
Human Memory Issues
    Account for human memory limitations by giving the user frequent closure on tasks
    Let the user recognize, rather than having to recall, whenever feasible

Cognitive Issues
    Use cognitive directness
    Draw on real-world analogies
Feedback
    Use informative feedback
    Give the user appropriate status indicators
System Messages
    Use user-centered wording in messages
    Use positive, nonthreatening wording in error messages
    Use specific, constructive terms in error messages
Anthropomorphism
    Do not anthropomorphize
Modality and Reversible Actions
    Use modes cautiously
    Make user actions easily reversible
Getting the User's Attention
    Get the user's attention judiciously
Display Issues
    Maintain display inertia
    Organize the screen to manage complexity
Individual User Differences
    Accommodate individual user experiences and differences
    Accommodate user experience levels
See also Neilsen, 1990 [Included], Hobart, 1995 [Included]

Analyzing User Errors
Input Errors
    Missed input
   Misinterpreted input
    Discarding (Filtering) of input
Processing Errors
    Sequence error in processing input
    Failure to process input
    Incorrect processing of input
    Partial processing of input
Response Errors
    Sequence error in response to input
    Failure to respond to input
    Inappropriate response to input
    Speed error in response to input
    Partial response to input

Low-fidelity prototyping
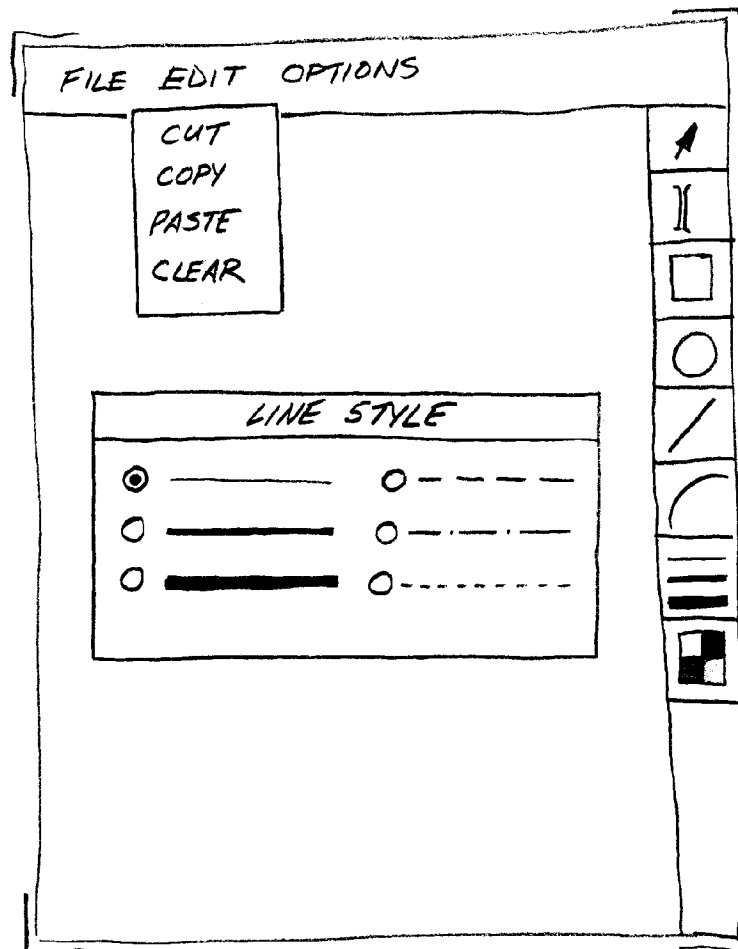
Rapid Prototyping

Prototypes don't have a lifetime. They are typically throw-away systems used only to study:
- the functionality of the system
- operation sequences
- user support needs
- required representations
- look and feel

Tools for Rapid Prototyping:
Macromedia Director (cross-platform)
GUI Builders:
    Hypercard, Supercard (MacOS)
    RapidApp (Irix)
    UIM/X (X-Windows)
    Nextstep (NEXTSTEP)
    C++/Views (MS Windows, cross-platform)
   Symantec Cafe Studio (Java, cross-platform)

Don't prototype unless:
- objectives and plans are well understood and acknowledged by both designers and users
- possible frequent changes in user requirements

User Interface Management Systems

Given a set of functional modules provided by an application specialist, a user interface designer builds a specialized interface for it. More than a toolkit, a UIMS provides support for:
- Syntax or sequences of commands or parameters
- Window management
- Help and documentation
- Screen format control
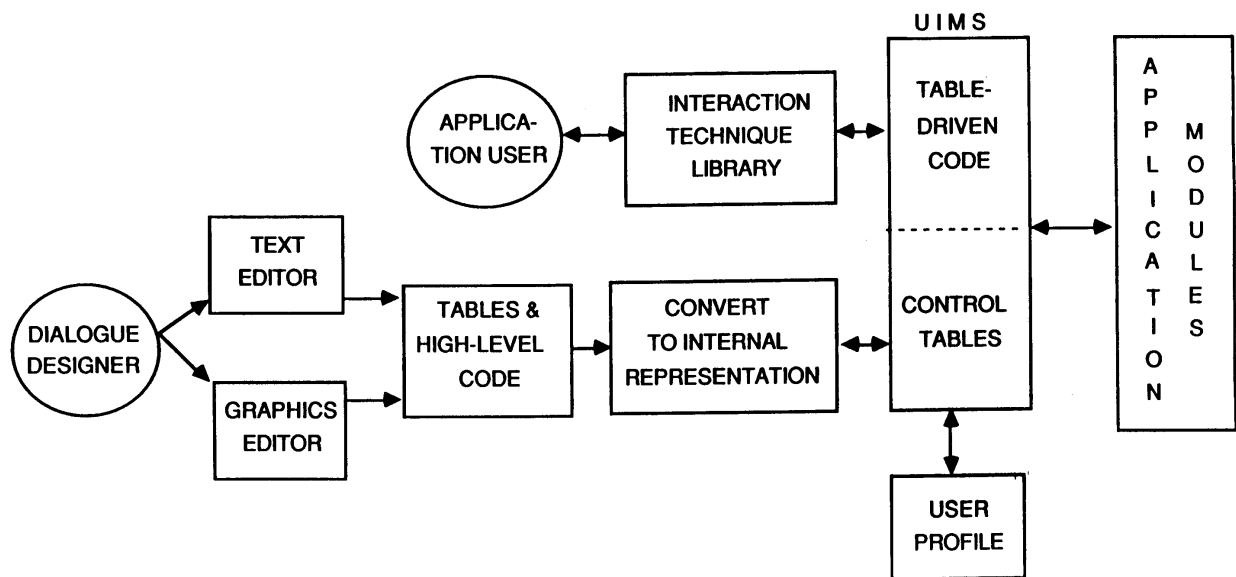
GUI Toolkits/Frameworks

What's the difference between a UIMS, a Toolkit, and a Framework?
(See insert).